WHAT IS CLAIMED IS:

1. A method for instrumenting software for use in an object-oriented environment, comprising:

receiving compiled application code;

instrumenting the compiled application code, instrumenting including adding code to turn features of instrumentation ON and OFF;

loading the instrumented code; and

executing the loaded code.

2. The method of claim 1, wherein instrumenting includes adding a collector object and a plurality of instrument data structure objects.

3. The method of claim 2, further comprising:

registering a list of methods in the collector object;

associating a selected one of the plurality of instrument data structure objects with a selected method in the list of methods; and

registering the selected instrument data structure object in the collector object.

4. The method of claim 1, wherein instrumenting is selective, based on a predetermined description.

5. The method of claim 4, wherein the predetermined description is indicative of at least one method to be instrumented.

6. The method of claim 4, wherein the predetermined description is indicative of at least one class to be instrumented.

7. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for instrumenting software, said method steps comprising:

receiving compiled application code;

instrumenting the compiled application code, instrumenting including adding code to turn features of instrumentation ON and OFF;

loading the instrumented code; and

executing the loaded code.

8. A software instrumentation architecture, comprising:

a plurality of methods configured to perform monitoring functions, each of the plurality of methods associated with an application object;

a plurality of instrument data objects, each of the plurality of methods associated with at least one of the plurality of instrument data objects, each of the plurality of instrument data objects configured to receive and store performance data from associated methods; and

a collector object coupled to the plurality of instrument data objects.

9. The architecture in claim 8, wherein the monitoring functions include recording a start time and recording an end time.

10. The architecture in claim 8, wherein each of the plurality of instrument data objects include a switch variable, the switch variable controlling the activation state for each of the associated methods.

11. The architecture in claim 10, the collector object including an instrument method for setting the switch variable in each of the plurality of instrument data structure objects.

12. The architecture in claim 8, wherein the collector object includes a list, the list including a representation of associations between the plurality of methods and the plurality of instrument data objects.

13. The architecture in claim 12, the collector object including an interface to a console, the console configured to display the list.

14. A method for software instrumentation, comprising:

receiving a state command for a class, the class associated with a plurality of methods;

selecting one of a plurality of instrument data structure objects based on an association between the plurality of methods and plurality of instrument data structure objects;

sending a state message to the selected one of a plurality of instrument data structures, the state message based on the state command; and

setting a switch variable based on the state message.

15. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for instrumenting software, said method steps comprising:

receiving a state command for a class, the class associated with a plurality of methods;

selecting one of a plurality of instrument data structure objects based on an association between the plurality of methods and plurality of instrument data structure objects;

sending a state message to the selected one of a plurality of instrument data structures, the state message based on the state command; and

setting a switch variable based on the state message.

16. A method for software instrumentation, comprising:

determining whether to perform a monitoring function by reading a switch variable;

setting an flag to TRUE if the determination is in the affirmative;

recording a start time if the flag is TRUE;

executing application code associated with the monitoring function; and

recording an end time if the flag is TRUE.

17. The method of claim 16, wherein the switch variable is stored externally, and the flag is stored internally, with respect to a module performing the method.

18. The method of claim 16, wherein each of recording a start time and recording an end time include making a call external to a module performing the method.

19. The method of claim 16, further comprising calculating a run time based on the recorded start time and the recorded end time.

20. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for instrumenting software, said method steps comprising:

determining whether to perform a monitoring function by reading a switch;

setting an flag to TRUE if the determination is in the affirmative;

recording a start time if the flag is TRUE;

executing application code associated with the monitoring function; and

recording an end time if the flag is TRUE.

21. A method for software instrumentation, comprising:

measuring a performance parameter;

comparing the measured performance parameter to a predetermined upper threshold; and

deactivating at least a portion of software instrumentation if the performance parameter is greater than the predetermined upper threshold.

22. The method of claim 21, further comprising:

comparing the measured performance parameter to a predetermined lower threshold; and

activating at least a portion of software instrumentation if the performance parameter is less than the predetermined lower threshold.